Data in Brief 31 (2020) 105829

Contents lists available at ScienceDirect

Data in Brief

journal homepage: www.elsevier.com/locate/dib

Data Article

Data stream dataset of SARS-CoV-2 genome

Raquel de M. Barbosa [a,b,d,**], Marcelo A.C. Fernandes [b,c,e,*,***]

[a] Laboratory of Drug Development, Department of Pharmacy, Federal University of Rio Grande do Norte, Natal, RN 59078-970, Brazil
[b] Laboratory of Machine Learning and Intelligent Instrumentation, IMD/nPITI, Federal University of Rio Grande do Norte, Natal 59078-970, Brazil
[c] Department of Computer Engineering and Automation, Federal University of Rio Grande do Norte, Natal, RN 59078-970, Brazil
[d] MIT Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA, 02142, USA
[e] John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138, USA

ARTICLE INFO

Article history:
Received 20 April 2020
Revised 2 June 2020
Accepted 4 June 2020
Available online 10 June 2020

Keywords:
SARS-CoV-2
Data stream
COVID-19

ABSTRACT

As of May 25, 2020, the novel coronavirus disease (called COVID-19) spread to more than 185 countries/regions with more than 348,000 deaths and more than 5,550,000 confirmed cases. In the bioinformatics area, one of the crucial points is the analysis of the virus nucleotide sequences using approaches such as data stream techniques and algorithms. However, to make feasible this approach, it is necessary to transform the nucleotide sequences string to numerical stream representation. Thus, the dataset provides four kinds of data stream representation (DSR) of SARS-CoV-2 virus nucleotide sequences. The dataset provides the DSR of 1557 instances of SARS-CoV-2 virus, 11540 other instances of other viruses from the Virus-Host DB dataset, and three instances of Riboviria viruses from NCBI (Betacoronavirus RaTG13, bat-SL-CoVZC45, and bat-SL-CoVZXC21).

* Corresponding author at: Department of Computer Engineering and Automation, Federal University of Rio Grande do Norte, Natal, RN, 59078-970, Brazil.
** Present address: MIT Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA, 02142, USA.
*** Present address: John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138, USA.
E-mail addresses: raquelmb@mit.edu (R.d.M. Barbosa), mfernandes@dca.ufrn.br (M.A.C. Fernandes).

https://doi.org/10.1016/j.dib.2020.105829

Specifications Table	
Subject	Biochemistry, Genetics and Molecular Biology (General)
Specific subject area	Bioinformatics
Type of data	Table Number
How data were acquired	NCBI - Genbank - SARS-CoV2 https://www.ncbi.nlm.nih.gov/genbank/sars-cov-2-seqs Virus-Host-DB https://www.genome.jp/virushostdb Matlab Software Excel Software
Data format	Raw and analyzed data are in Matlab file (.mat) and Microsoft Excel file (.xlsx).
Parameters for data collection	The entire dataset was generated using MATLAB 2019b on Windows operating system with Intel Core - i5 6500T 2.5 GHz quad-core processor with 16GB of RAM.
Description of data collection	The raw data were downloaded from NCBI - Genbank, and Virus-Host-DB. The data stream values were generated using Matlab.
Data source location	Laboratory of Machine Learning and Intelligent Instrumentation, IMD/nPITI, Federal University of Rio Grande do Norte.
Data accessibility	https://data.mendeley.com/datasets/g5ktw4y4pz/2

Value of the Data

- These data are useful because they provide numeric representation of the COVID-2019 epidemic virus (SARS-CoV-2). With this, it is possible to use data stream algorithms.
- All researchers in bioinformatics, computing science, and computing engineering disciplines can benefit from these data because by using this numeric representation, they can apply several stream algorithms and techniques such as TEDA (Typicality and Eccentricity Data Analytic), TEDA-Cloud, TEDA-Cluster and Teda-Class in genomic information.
- Data experiments that use analytic stream techniques in SARS-CoV-2 virus genomic information can be used with this dataset.
- These data represent an simple way to evaluate the SARS-CoV-2 virus genome with stream algorithms.
- Differently of the conventional bioinformatics techniques in which are based on dynamic programming (such as BLAST and other), this approach allows the utilization of different techniques (techniques commons in other areas) to find similarities between genome sequences.

1. Data Description

This work presents a dataset of data stream representation (DSR) of SARS-CoV-2 virus nucleotide sequences. The dataset contains two kinds of data, the raw data, and the processing data. The raw data is composed of the 1557 instances of the SARS-CoV-2 virus genome collected from the National Center for Biotechnology Information (NCBI) [1], 11540 instances of other viruses from the Virus-Host DB [2,3], and the other three specific viruses also collected from NCBI (Betacoronavirus RaTG13, bat-SL-CoVZC45, and bat-SL-CoVZXC21). The last specific three viruses have high similarity with SARS-CoV-2 [4,5]. The processing data is composed of four kinds of DSR called Direct Mapping (DM), DM with Chaos Game Representation (DM-CGR), *k*-mers mapping (kMersM) and *k*-mers mapping with CGR (kMersM-CGR). *k*-mers is a frequency count metric used in Bioinformatics. Other *k*-mers datasets are presented in [6–8].

In the Chaos Game Representation (CGR) [8], the genome sequence is transformed in a bi-dimensional signal (1D vector), and after that, this signal passes to infinite impulse response (IIR) filter [9]. The result of CGR is a signal that expressed the density of the bases and, at the same time, the transition between bases because the IIR is a memory system. CGR can be used

with the signature of the genome sequence. With k-mers representation [10], the genome can be transformed into a 1D or 2D vector that represents the occurrence number of each base (frequency of the bases). k-mers also can be used with a signature of the genome sequence. However, in this manuscript, the genome sequence is transformed into a linear stream data, and this type of transformation can be used with stream algorithms. Another important aspect of this dataset is associated with applied CGR not in all sequences but just in each k bases (with mers or not). This strategy maintains the statistical characteristics and reduces the size of the stream.

The data is organized into three main directories: "SARS-CoV-2 data", "Virus-Host DB data" and "Other viruses data". Each main directory contains three files called "RawDataTable.mat", "RawData.mat" and "RawData.xlsx", and four sub-directories named "DirectMapping", "DirectMappingCGR", "kmersMapping" and "kmersMappingCGR". "RawDataTable.mat", "RawData.mat" and "RawData.xlsx" files store the raw data information from viruses database; they have the same information, however in the "RawDataTable.mat" the attributes are stored in Matlab table format (after 2013b version), in the "RawData.mat" the attributes are stored in Matlab cell arrays format, and in the "RawData.xlsx" the attributes are stored in a Microsoft Excel file. In the sub-directories "DirectMapping", "DirectMappingCGR", "kmersMapping" and "kmersMappingCGR" are stored the DM, DM-CGR, kMersM and kMersM-CGR data stream representation, respectively. Inside each sub-directory the files are called:

- For DM, the DSR was generated for $k = 1 \ldots 5$ and the files are called "PointsData_1_k=***k***.mat";
- For DM-CGR, the DSR was generated for $k = 1 \ldots 7$ and the files are called "PointsDataCGR_1_k=***k***.mat";
- For kMersM, the DSR was generated for $k = 2 \ldots 5$ and the files are called "PointsDatakmers_1_k=***k***.mat";
- For kMersM-CGR:
  - In the directories "Other viruses data" and "SARS-CoV-2 data", the DSR was generated for $k = 2 \ldots 7$ and the files are called "PointsDatakmersCGR_1_k=***k***.mat";
  - In the "Virus-Host DB data", the DSR was generated for $k = 2, 3, 5,$ and 7 and the files are called "PointsDatakmersCGR_1_k=***k***.mat";

For the main directory "Virus-Host DB data", the values are stored in 10 files where each i-th file is called "PointsData_***k***_k=***k***.mat" for sub-directory "DirectMapping", "PointsDataCGR_***i***_k=***k***.mat" for DM-CGR, "PointsDatakmers_***i***_k=***k***.mat" for kMersM and "PointsDatakmersCGR_***i***_k=***k***.mat" for kMersM-CGR.

2. Experimental design, materials, and methods

The streams were based in nucleotide sequence, **s**, expressed as

$$\mathbf{s} = [s_1, \ldots, s_n, \ldots, s_N] \tag{1}$$

where N is the length of sequence and s_n is the nth nucleotide of the sequence.

For DM and DM-CGR, the nucleotide sequence, **s**, are grouped in sub-sequences of the k bases. The group of sub-sequences can be expressed as

$$\mathbf{B} = \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_i \\ \vdots \\ \mathbf{b}_K \end{bmatrix} = \begin{bmatrix} s_1 & \cdots & s_k \\ \vdots & \ddots & \vdots \\ s_{k(i-1)+1} & \cdots & s_{k(i-1)+k} \\ \vdots & \ddots & \vdots \\ s_{K-k+1} & \cdots & s_K \end{bmatrix} \tag{2}$$

where

$$K = k \times \left\lfloor \frac{N}{k} \right\rfloor \tag{3}$$

and the i-th vector $\mathbf{b}_i$ is a i-th group of the k nucleotides, that is

$$\mathbf{b}_i = \left[b_{i,1}, \ldots, b_{i,j}, \ldots, b_{i,k}\right] = \left[s_{k(i-1)+1}, \ldots, s_{k(i-1)+j}, \ldots, s_{k(i-1)+k}\right]. \tag{4}$$

For DM, the group of sup-sequences, stored in matrix $\mathbf{B}$, are transformed in a sequence of the integer values expressed as

$$\mathbf{c} = [c_1, \ldots, c_i, \ldots, c_K] \tag{5}$$

where $\mathbf{c}$ is the DM stream stored in dataset. The DM stream, $\mathbf{c}$, calculus can be expressed as

$$\begin{bmatrix} c_1 \\ \vdots \\ c_i \\ \vdots \\ c_K \end{bmatrix}^T = f_{\text{map}}(\mathbf{B}) = \begin{bmatrix} f_{\text{map}}(\mathbf{b}_1) \\ \vdots \\ f_{\text{map}}(\mathbf{b}_i) \\ \vdots \\ f_{\text{map}}(\mathbf{b}_K) \end{bmatrix} \tag{6}$$

where $f_{\text{map}}(\cdot)$ is the mapping function expressed by

$$c_i = f_{\text{map}}(\mathbf{b}_i) = \left(\sum_{j=0}^{k-1} 4^j \times \left(u_{i,j} - 1\right)\right) + 1 \tag{7}$$

and

$$u_{i,j} = \begin{cases} 1 & \text{for } b_{i,j+1} = \text{T or U} \\ 2 & \text{for } b_{i,j+1} = \text{C} \\ 3 & \text{for } b_{i,j+1} = \text{A} \\ 4 & \text{for } b_{i,j+1} = \text{G} \end{cases}. \tag{8}$$

For DM-CGR, the stream is characterized by vector $\mathbf{a}$ expressed as

$$\mathbf{a} = [a_1, \ldots, a_i, \ldots, a_K] \tag{9}$$

where the a_i is the i-th value of CGR. In CGR (see [11,12]) each element a_i is a bi-dimensional value expressed as

$$a_i = (a_i^x, a_i^y) \tag{10}$$

where a_i^x and a_i^y are the x-axes and y-axes in bi-dimensional space, receptively. The values of the CGR are calculate using the functions $f_{\text{CGR}}^x(\cdot)$ and $f_{\text{CGR}}^y(\cdot)$ in Matrix $\mathbf{B}$, that is

$$\begin{bmatrix} (a_1^x, a_1^y) \\ \vdots \\ (a_i^x, a_i^y) \\ \vdots \\ (a_K^x, a_K^y) \end{bmatrix}^T = \left(f_{\text{CGR}}^x(\mathbf{B}), f_{\text{CGR}}^y(\mathbf{B})\right) = \begin{bmatrix} \left(f_{\text{CGR}}^x(\mathbf{b}_1), f_{\text{CGR}}^y(\mathbf{b}_1)\right) \\ \vdots \\ \left(f_{\text{CGR}}^x(\mathbf{b}_i), f_{\text{CGR}}^y(\mathbf{b}_i)\right) \\ \vdots \\ \left(f_{\text{CGR}}^x(\mathbf{b}_K), f_{\text{CGR}}^y(\mathbf{b}_K)\right) \end{bmatrix}. \tag{11}$$

The function $f_{\text{CGR}}^x(\cdot)$ calculates the x-axes value of the CGR and it can be expressed as

$$a_i^x = f_{\text{CGR}}^x(\mathbf{b}_i) = p_{i,k}^x \tag{12}$$

where

$$p_{i,j}^x = \frac{1}{2} u_{i,j}^x + \frac{1}{2} p_{i,j-1}^x, \text{ for } j = 1, \ldots, k \tag{13}$$

and

$$u_{i,j}^x = \begin{cases} 1 & \text{for } b_{i,j} = \text{A} \\ -1 & \text{for } b_{i,j} = \text{T or U} \\ -1 & \text{for } b_{i,j} = \text{C} \\ 1 & \text{for } b_{i,j} = \text{G} \end{cases}. \tag{14}$$

For y-axes, the function, $f^{y}_{\mathrm{CGR}}(\cdot)$, can be expressed as

$$a^{y}_{i} = f^{y}_{\mathrm{CGR}}(\mathbf{b}_i) = p^{y}_{i,k} \tag{15}$$

where

$$p^{y}_{i,j} = \frac{1}{2}u^{y}_{i,j} + \frac{1}{2}p^{y}_{i,j-1}, \text{ for } j = 1, \ldots, k \tag{16}$$

and

$$u^{y}_{i,j} = \begin{cases} 1 & \text{for } b_{i,j} = \mathrm{A} \\ 1 & \text{for } b_{i,j} = \mathrm{T} \text{ or } \mathrm{U} \\ -1 & \text{for } b_{i,j} = \mathrm{C} \\ -1 & \text{for } b_{i,j} = \mathrm{G} \end{cases}. \tag{17}$$

For the initial condition, $j = 0$, $p^{x}_{i,0} = \alpha_x$ and $p^{y}_{i,0} = \alpha_y$ [11,12]. The dataset was generated with $\alpha_x = 0$ and $\alpha_y = 0$.

For kMersM and kMersM-CGR, the nucleotide sequence, **s**, are grouped in k-mers subsequences [13,14] in the matrix **H** that can expressed as

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_i \\ \vdots \\ \mathbf{h}_{N-k} \\ \mathbf{h}_{N-k+1} \end{bmatrix} = \begin{bmatrix} s_1 & \cdots & s_k \\ s_2 & \cdots & s_{k+1} \\ \vdots & \ddots & \vdots \\ s_i & \cdots & s_{i+k} \\ \vdots & \ddots & \vdots \\ s_{N-k} & \cdots & s_{N-1} \\ s_{N-k+1} & \cdots & s_N \end{bmatrix}. \tag{18}$$

The kMersM, stream is characterized as a sequence of the integer values expressed as

$$\mathbf{r} = \left[r_1, \ldots, r_i, \ldots, r_{N-k+1}\right] \tag{19}$$

where

$$\begin{bmatrix} r_1 \\ \vdots \\ r_i \\ \vdots \\ h_{N-k+1} \end{bmatrix}^{T} = f_{\mathrm{map}}(\mathbf{H}) = \begin{bmatrix} f_{\mathrm{map}}(\mathbf{h}_1) \\ \vdots \\ f_{\mathrm{map}}(\mathbf{h}_i) \\ \vdots \\ f_{\mathrm{map}}\left(\mathbf{h}_{N-k+1}\right) \end{bmatrix}. \tag{20}$$

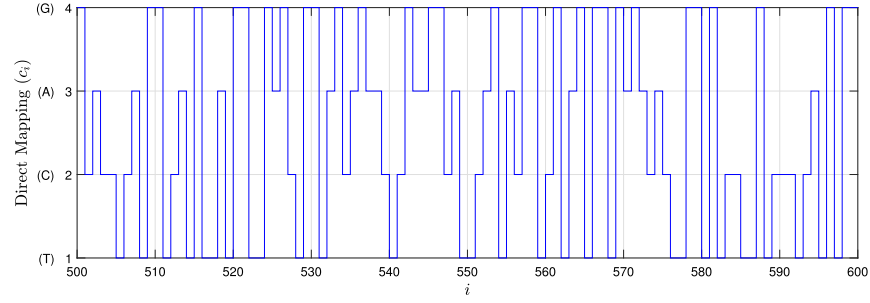
The function $f_{\mathrm{map}}(\cdot)$ is the mapping processing characterized by Eqs. (7) and (8). The kMersM-CGR is stored in the vector **z** expressed as

$$\mathbf{z} = \left[z_1, \ldots, z_i, \ldots, z_{N-k+1}\right] \tag{21}$$

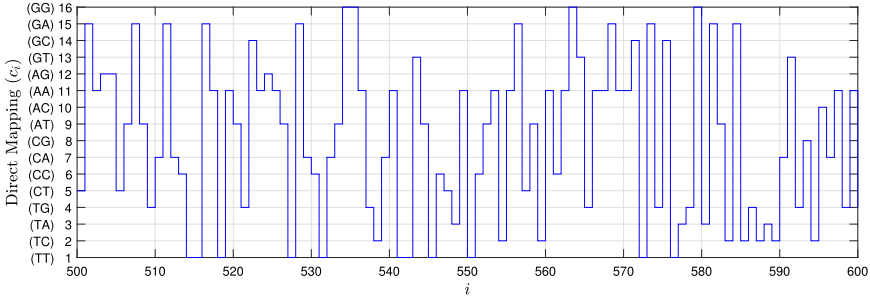
where the z_i is the i-th value of CGR. Each ith element z_i is a bi-dimensional value expressed as

$$z_i = (z^{x}_{i}, z^{y}_{i}) \tag{22}$$

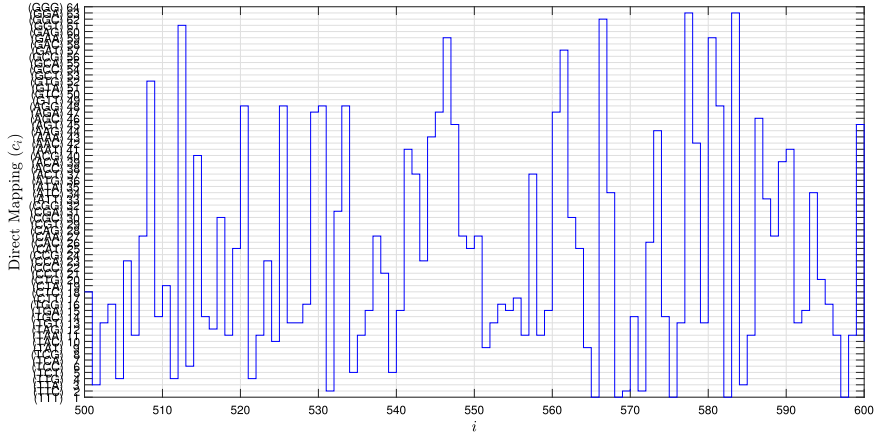
where z^{x}_{i} and z^{y}_{i} are the x-axes and y-axes in bi-dimensional space, receptively. The values of the CGR are calculate using the functions $f^{x}_{\mathrm{CGR}}(\cdot)$ (see Eqs. (12)–(14)) and $f^{y}_{\mathrm{CGR}}(\cdot)$ (see Equation



(a) DM-DSR for $k = 1$

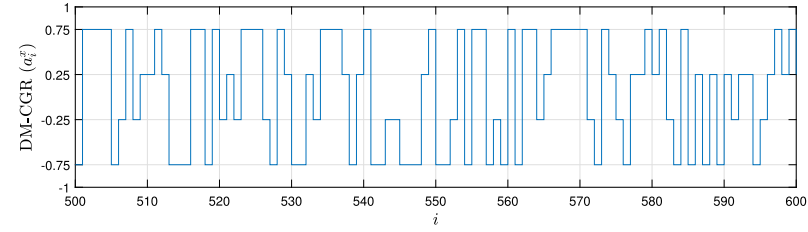


(b) DM-DSR for $k = 2$.

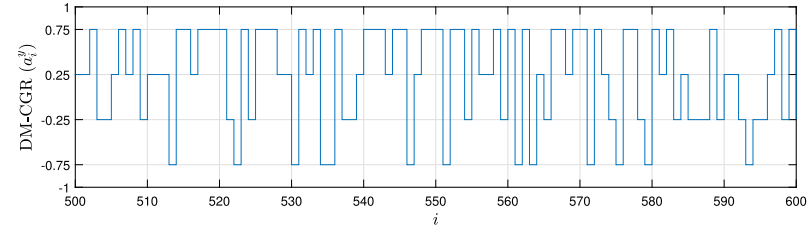


(c) DM-DSR for $k = 3$.

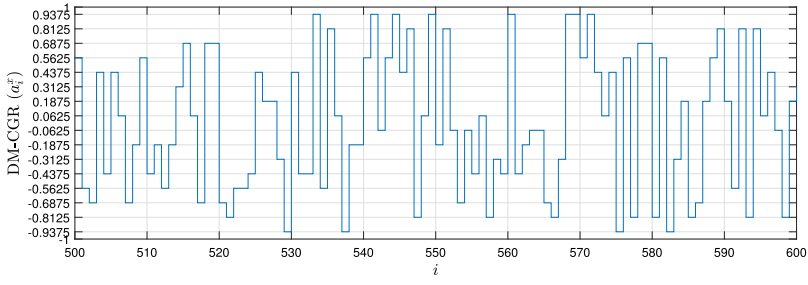
Fig. 1. Example of the DM-DSR values for the SARS-CoV-2 sequence ($i = 500 \ldots 600$) stored in dataset (MT126808 - Brazil).



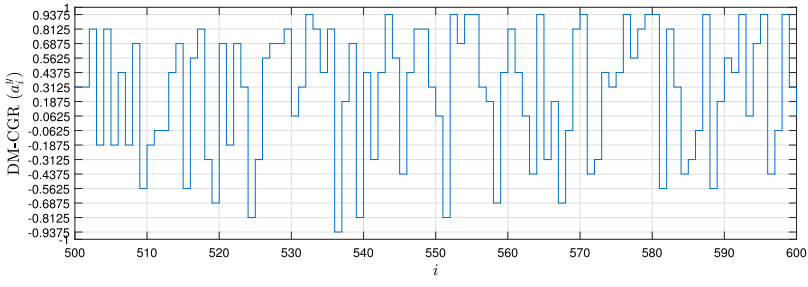
(a) DM-CGR-DSR a_i^x for $k = 2$.



(b) DM-CGR-DSR a_i^y for $k = 2$.

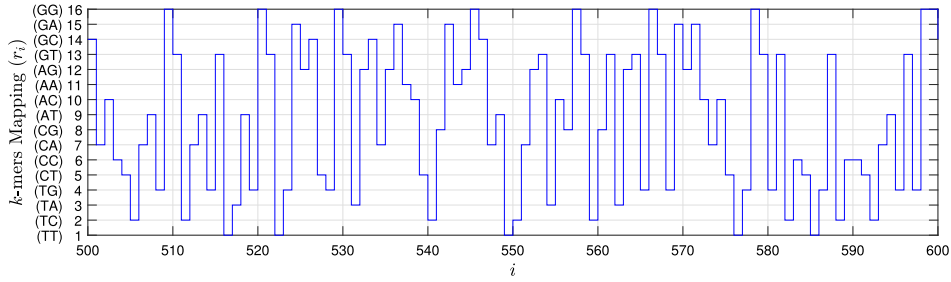


(c) DM-CGR-DSR a_i^x for $k = 4$.

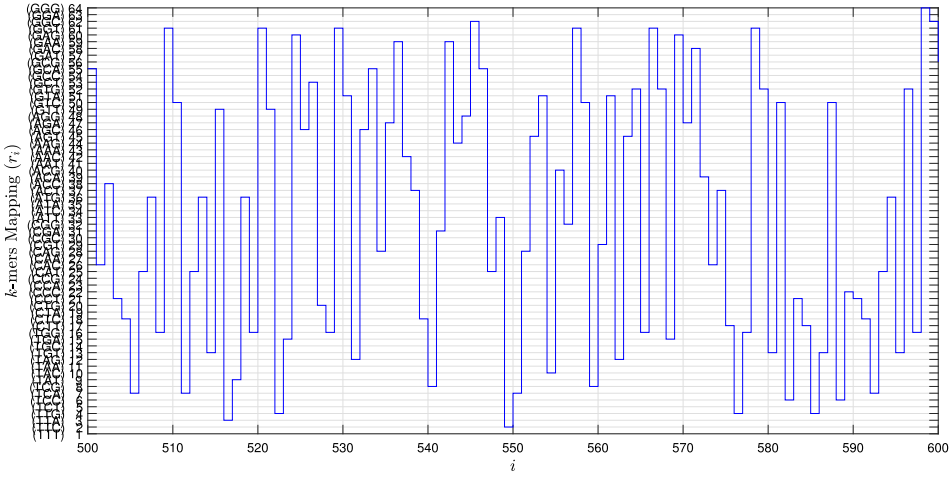


(d) DM-CGR-DSR a_i^y for $k = 4$.

Fig. 2. Example of the DM-CGR-DSR values for the SARS-CoV-2 sequence ($i = 500 \ldots 600$) stored in dataset (MT126808 - Brazil).



(a) kMersM-DSR for $k = 2$



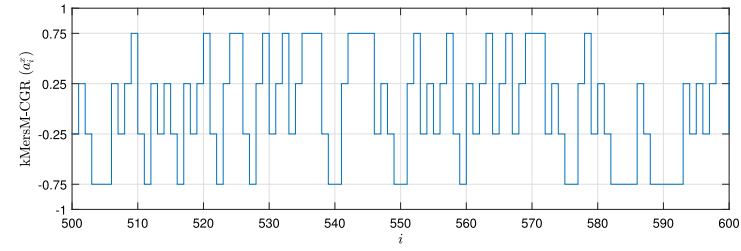
(b) kMersM-DSR for $k = 3$.

Fig. 3. Example of the kMersM-DSR values for the SARS-CoV-2 sequence ($i = 500 \ldots 600$) stored in dataset (MT126808 - Brazil).

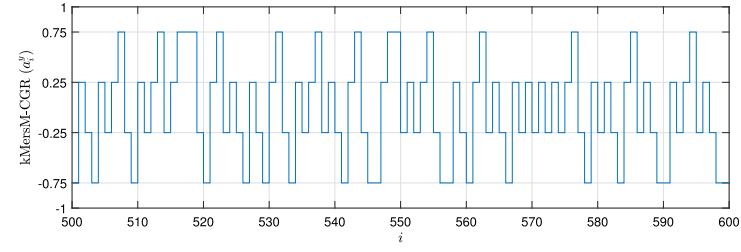
see Eqs. (15)–(17)) in Matrix **H**, that is

$$\begin{aligned}\begin{bmatrix} (z_1^x, z_1^y) \\ \vdots \\ (z_i^x, z_i^y) \\ \vdots \\ (z_{N-k+1}^x, z_{N-k+1}^y) \end{bmatrix}^T &= \left(f_{\text{CGR}}^x(\mathbf{H}), f_{\text{CGR}}^y(\mathbf{H})\right) \\ &= \begin{bmatrix} \left(f_{\text{CGR}}^x(\mathbf{h}_1), f_{\text{CGR}}^y(\mathbf{h}_1)\right) \\ \vdots \\ \left(f_{\text{CGR}}^x(\mathbf{h}_i), f_{\text{CGR}}^y(\mathbf{h}_i)\right) \\ \vdots \\ \left(f_{\text{CGR}}^x(\mathbf{h}_{N-k+1}), f_{\text{CGR}}^y(\mathbf{h}_{N-k+1})\right) \end{bmatrix}.\end{aligned} \tag{23}$$

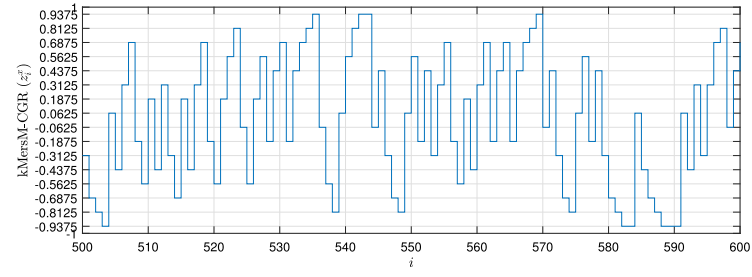
Figs. 1–4 show the DSR examples for SARS-CoV-2 from Brazil, respectively.



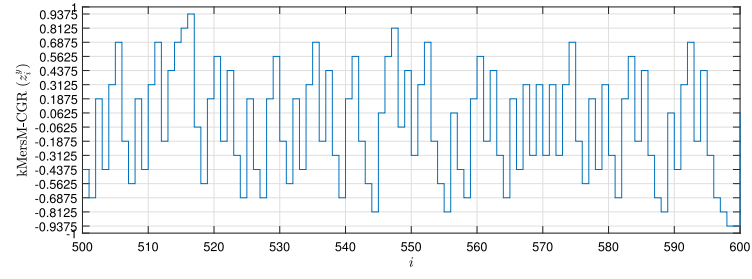
(a) kMersM-CGR-DSR a_i^x for $k = 2$.



(b) kMersM-CGR-DSR a_i^y for $k = 2$.



(c) kMersM-CGR-DSR a_i^x for $k = 4$.



(d) kMersM-CGR-DSR a_i^y for $k = 4$.

Fig. 4. Example of the kMersM-CGR-DSR values for the SARS-CoV-2 sequence ($i = 500 \ldots 600$) stored in dataset (MT126808 - Brazil).

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors wish to acknowledge the financial support of the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) for their financial support.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.dib.2020.105829

References

[1] NCBI, SARS-CoV-2 (Severe acute respiratory syndrome coronavirus 2) Sequences, 2020, (https://www.ncbi.nlm.nih.gov/genbank/sars-cov-2-seqs/).
[2] T. Mihara, Y. Nishimura, Y. Shimizu, H. Nishiyama, G. Yoshikawa, H. Uehara, P. Hingamp, S. Goto, H. Ogata, Linking virus genomes with host taxonomy, Viruses 8 (3) (2016), doi:10.3390/v8030066. URL https://www.mdpi.com/1999-4915/8/3/66
[3] Virus-Host DB, Virus-Host DB - Website, 2020, https://www.genome.jp/virushostdb.
[4] G.S. Randhawa, M.P. Soltysiak, H.E. Roz, C.P. de Souza, K.A. Hill, L. Kari, Machine learning using intrinsic genomic signatures for rapid classification of novel pathogens: Covid-19 case study, bioRxiv (2020a), doi:10.1101/2020.02.03.932350.
[5] G.S. Randhawa, M.P. Soltysiak, H.E. Roz, C.P. de Souza, K.A. Hill, L. Kari, Machine learning-based analysis of genomes suggests associations between wuhan 2019-ncov and bat betacoronaviruses, bioRxiv (2020b), doi:10.1101/2020.02.03.932350.
[6] R. de M. Barbosa, M.A. Fernandes, Chaos game representation dataset of sars-cov-2 genome, Mendeley Data v2 (2020a), doi:10.17632/nvk5bf3m2f.2.
[7] R. de M. Barbosa, M.A. Fernandes, k-mers 1d and 2d representation dataset of sars-cov-2 nucleotide sequences, Mendeley Data v2 (2020b), doi:10.17632/f5y9cggnxy.2.
[8] R. de M. Barbosa, M.A. Fernandes, Chaos game representation dataset of sars-cov-2 genome, Data Brief 30 (2020c) 105618, doi:10.1016/j.dib.2020.105618.
[9] J.G. Proakis, D.K. Manolakis, Digital Signal Processing, (4th Ed.), Prentice-Hall, Inc., USA, 2006.
[10] L. Pinello, G. Lo Bosco, G.-C. Yuan, Applications of alignment-free methods in epigenomics, Briefings in Bioinf. 15 (3) (2014) 419–430.
[11] H. Jeffrey, Chaos game representation of gene structure, Nucleic Acids Research 18 (8) (1990) 2163–2170, doi:10.1093/nar/18.8.2163.
[12] C. Yin, Encoding dna sequences by integer chaos game representation, 2017, arXiv: 1712.04546
[13] D. Mapleson, G. Garcia Accinelli, G. Kettleborough, J. Wright, B.J. Clavijo, KAT: a K-mer analysis toolkit to quality control NGS datasets and genome assemblies, Bioinformatics 33 (4) (2016) 574–576, doi:10.1093/bioinformatics/btw663. URL https://academic.oup.com/bioinformatics/article-pdf/33/4/574/25146635/btw663.pdf
[14] B. Chor, D. Horn, N. Goldman, Y. Levy, T. Massingham, Genomic dna k-mer spectra: models and modalities, Genome Biol. 10 (10) (2009) R108.